Intel® Wired for Management Baseline Specification Self-Test Instruction Manual

Version 2.0

Self-Test Plan for Wired for Management Baseline Specification Version 2.0 Platforms: Desktop, Mobile and Server

February 7, 2000
Intel Corporation

“Year 2000 Capable”
An Intel® product, when used in accordance with its associated documentation, is “Year 2000 Capable” when, upon installation, it accurately stores, displays, processes, provides, and/or receives date data from, into, and between the 20th and 21st centuries, including leap year calculations, provided that all other technology used in combination with said product properly exchanges date data with it.

Table of Contents

Section 1

Introduction

Intended Audience

This document provides general instructions for the use of the tools contained in the Wired for Management (WfM) Baseline Specification Self-Test ToolKit v2.0. The goal of the document is to provide step-by-step instructions for testing features supported by the WfM Baseline Specification v2.0 on Original Equipment Manufacturer (OEM) systems.

About this Document

This document is a general instruction manual on how to test Wired for Management Baseline v2.0 platforms using the tools contained in the Wired for Management Baseline Specification Self-Test ToolKit v2.0. This document does not replace the documentation included with each of the tools. Refer to the individual tools' documentation for detailed installation notes, instructions on use and troubleshooting guides.

Purpose of Tools

The programs included in the ToolKit test the basic functionality of the WfM Baseline Specification v2.0 from an external test perspective. These tools are provided as an aid to developers of WfM-capable platforms and add-ins, and are intended to provide a high level of confidence that the WfM-capable product has been implemented correctly. **Passing these tests does not imply any certification or guarantee of any specific functionality or interoperability.**

Installation Notes

Please completely read "WfM Self-Test v2.0Setup" prior to installing the WfM Self-Test ToolKit v2.0. This section provides information on how to properly install the tools based on the network being used for testing.

Install the WfM Self-Test ToolKit v2.0 according to the test network being used. After the installation, read the documentation included with each individual tool. The remainder of this document refers to several sections contained within the individual tools' documentation.

Relevant Documents

- Wired for Management Baseline Specification Version 2.0 (http://developer.intel.com/ial/wfm/wfmspecs.htm)
- DMTF DMI v2.0s Specification (http://www.dmtf.org/spec/dmis.html)
- Wired for Management Design information (http://developer.intel.com/ial/wfm/)

Section
2

WfM Self-Test v2.0 Setup

Before installing the WfM Self-Test ToolKit v2.0, selecting a test network configuration is required. After a configuration is selected, the WfM Self-Test ToolKit v2.0 can be properly installed. The installation procedure prompts for information regarding the presence of Dynamic Host Configuration Protocol (DHCP) services on the server where the tools are being installed. The following configuration options contain diagrams that help determine the answers to this prompt.

NOTE: It is recommended that the test network be isolated from any production network and machines.

Option #1: Recommended Network Configuration

Configuration Elements

- A server platform running Windows* NT* Server (version 4.0 or higher) to install the Self-Test Toolkit onto. SNMP Services must be installed with a Community Name of “public”. This server is labeled “WfM Test Server” in Figure 1.
- A server platform running Windows NT (server OS version 4.0 or higher) with DHCP services enabled and configured with a user-defined DHCP address scope. This server is labeled “DHCP Server” in Figure 1.
- A Network Hub to provide connectivity between the network components.
- The system to be tested for WfM feature support, System Under Test (SUT). This is either a Preboot Execution Environment (PXE) Client, or a PXE Bootserver.

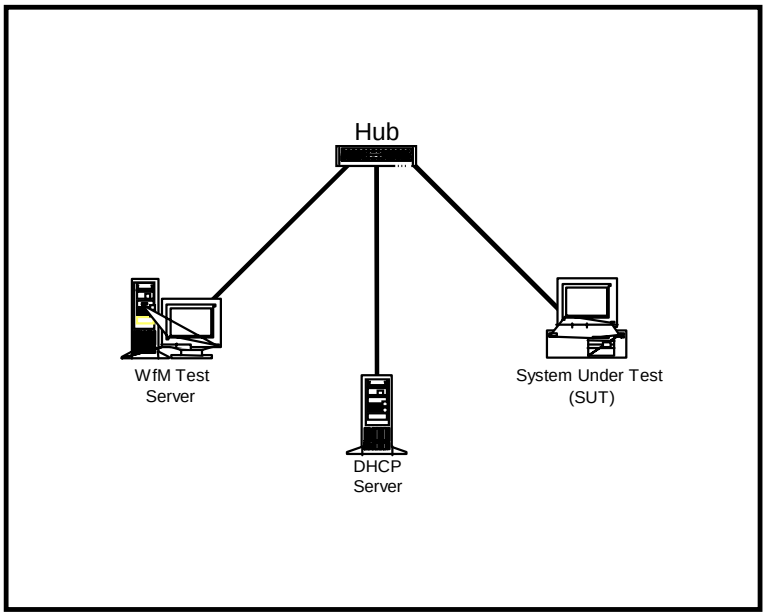

Figure 1. Recommended Network Configuration for Client Test

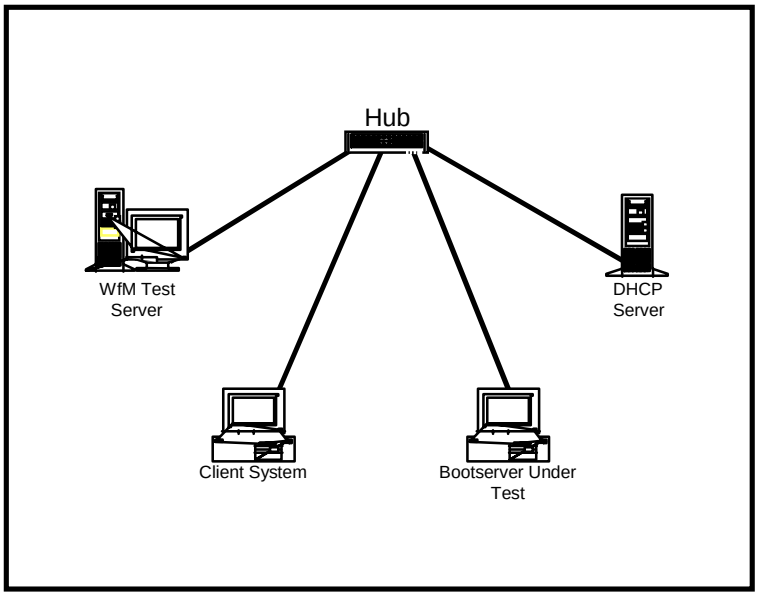

Figure 2. Recommended Network Configuration for Bootserver Test

Configuration Requirements

Ensure that the following requirements are met **prior** to installing the WfM Self-Test ToolKit v2.0:

- Windows NT (version 4.0 or higher) server with SNMP Services installed on the WfM Test Server.

- The Windows NT version 4.0 distribution CD-ROM is available. The Preboot Execution Environment (PXE) PDK needs files from this CD to install.

- Windows NT (version 4.0 or higher) server is installed on the DHCP Server. This server must be running Microsoft DHCP service with a defined DHCP address scope.

- Check the DHCP Server. Boot a DHCP client (this can be the System Under Test) and verify that the DHCP Server supplies an IP address to the client.

- Verify that the "GUEST" account is enabled.

Installation Requirements

After the DHCP Server has been verified, proceed with the installation of the WfM Self-Test ToolKit v2.0 on the WfM Test Server. When prompted during PXE PDK installation, "Is this going to be installed on a machine running DHCP Server?" , click No. Complete the installation as prompted, reading all provided documentation for the individual tools.

Option #2: Optional Network Configuration

Configuration Elements

- A server platform running a Windows NT (version 4.0 or higher) server OS with DHCP services enabled and configured with a user-defined DHCP address scope. Additionally, this server will host the WfM v2.0 Self-Test ToolKit, so SNMP Services must be installed prior to WfM v2.0 Self-Test ToolKit installation. This server is labeled "WfM Test/DHCP Server" in Figure 3.

- A Network Hub to provide connectivity between the network components.

- The system to be tested for WfM feature support (SUT). This will be either a PXE Client, or a PXE Bootserver.

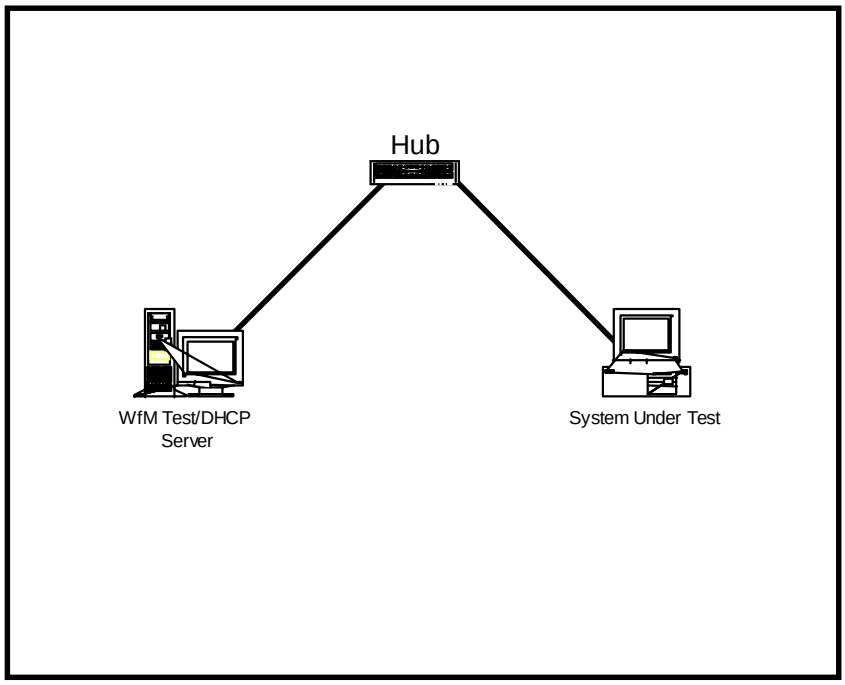

Figure 3. Optional Network Configuration for Client Test

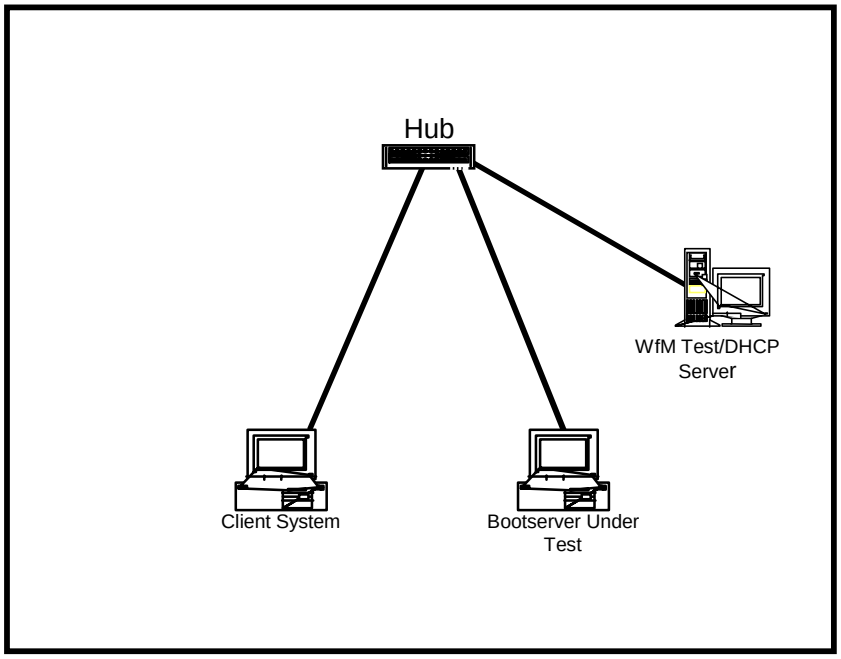

Figure 4. Optional Network Configuration for Bootserver Test

Configuration Requirements

Ensure that the following requirements are met **prior** to installing the WfM v2.0 Self-Test Tools:

- Windows NT (version 4.0 or higher) server is installed on the WfM Test/DHCP Server. Additionally, this server must be running a Microsoft DHCP service and a DHCP address scope must be defined. Install SNMP Services prior to the WfM v2.0 Self-Test ToolKit installation, setting Community Name to “public”.

- Using the same host to provide the DHCP service and to be the test server requires that a DHCP Class Identifier option, tag value 60, must be added to the DHCP service. This option must be set to “PXEClient”. Do not add this tag if the DHCP service is provided on a host other than the test server.

- Check the DHCP Server. Boot a DHCP client (this can be the System Under Test) and verify that the DHCP Server supplies an IP address to the client.

- Verify that the “GUEST” account is enabled.

Installation Requirements

After the DHCP Service has been verified, proceed with the installation of the WfM v2.0 Self-Test ToolKit on the WfM Test/DHCP Server. When prompted during PXE PDK installation, “Is this going to be installed on a machine running DHCP Server?” answer Yes. Complete the installation as prompted, read all documentation provided for the individual tools, and continue reading the information below.

WfM Test Server and WfM Test/DHCP Server Configuration

Using the PXE Configuration Utility (More information on this utility can be found in the PXE Product Development Kit Instructions, Section 4 “PXE Services Configuration”), configure proxyDHCP as follows:

Configuration common to Client and Bootserver Tests

1) Enter the Multicast Discovery listening address (e.g. 224.0.1.2) (Section 4.3).

2) Enable the Multicast Discovery method (Section 4.3).

3) Disable the Broadcast Discovery method (Section 4.3.).

Configuration specific to Bootserver Tests

4) Add the Bootserver under test in the Client Boot Menu. If necessary, define the Bootserver type in this step (Section 4.3 Client Boot Menu).

5) Enter the Multicast Discovery listening address (e.g. 224.0.1.2) in the Bootserver under test. (OEM specific - use the Bootserver manufacturer’s configuration instructions)

Pre-Test Checklist

Please verify that each of the items detailed in this checklist have been accomplished prior to testing the WfM client:

- Ensure that the test network is configured according to one of the topologies discussed in "WfM v2.0 Self-Test Setup".

- Verify that the DHCP server has been configured to provide IP addresses to network clients. To do this, boot a network client to its operating system and verify that it has received an IP address. Instructions for locating the IP address of the System Under Test (SUT) can be found in the documentation provided with the OS that is installed on the client. If the SUT is running Windows NT, you can find the IP address by running IPCONFIG from a DOS shell. If the SUT is running Windows 95 or Windows 98, go to the Start menu, click Run, type in WINIPCFG, and click OK. For more information on configuring a DHCP server see the PXE Product Development Kit Instructions, Section 3.2. "Set up the NT Server(s)".

- Verify that the Guest account is enabled on the PXE server.

- Verify that the DOS image files required to execute the PXETest have been created. The procedure for generating these image files is described in Section 3.4.1 of the PXE Product Development Kit Instructions (Create the boot files APITEST.1 and DOSUNDI.1).

Section
3

Power Management Testing

Power Management Requirements and Coverage

Requirements	Desktop	Mobile	Server	Self-Test Tool
ACPI-Compliant	**Required**	**Required**	**Required as defined in [HDG]**	**ISACPI**
Recommended Sleep Mode	**S3**	**S3**	**S1**	**N/A**
Remote Wake-up • Magic Packet* • Packet Filtering • Wake On-Ring	**Required** • **Required** • **Recommended** • **Recommended**	**Recommended**	**Recommended**	**RWUTool Supports Magic Packet and Packet Filtering**
Bus Power Management	**Required**	**Required**	**Recommended**	**N/A**

ACPI Testing

Overview of ACPI Test Tool

The Advanced Configuration and Power Interface (ACPI) test tool ISACPI.EXE provides the following features:

- The ability to detect the presence of ACPI support on the SUT.

Full ACPI and Bus Power Management testing is beyond the scope of this Self-Test ToolKit. For more information on implementing and testing ACPI visit the following web sites:

- PC99 Test Specification and the System Test Implementers Forum: http://www.systemtest.org
- ACPI Promoters: http://www.teleport.com/~acpi/

- Intel® Instantly Available PC Technology: http://developer.intel.com/solutions/tech/power.htm

- Microsoft Hardware Developer web site: http://microsoft.com/hwdev/

ACPI Test Procedure

1) Boot the SUT to DOS.

2) Run the ISACPI.EXE utility. The location is (c:\Program Files\Intel\WfM_Self_Test\ACPITool). It may also be found under the (c:\Program Files\Intel\WfM_Self_Test\Boot Disk) directory.

3) Record the result output on the screen.

Interpreting the Results of the ACPI Test

The ISACPI.EXE utility can detect the presence of ACPI support in the BIOS and report that ACPI is present on the system or not.

NOTE: Messages indicating the failure to open test.ini can be ignored.

RWUTool

Overview of Remote Wake-Up Test Tool

The Remote Wake-Up test tool provides the following features:

- The ability to send an Internet Control Message Protocol (ICMP) “ping” to the SUT to verify connectivity.

- The ability to send the SUT a properly formatted wake-up packet.

Remote Wake-Up Test Procedure

4) Boot the SUT to the OS.

5) Find the IP address and MAC address of the SUT. You can find instructions for locating the IP and MAC addresses of the SUT in the documentation provided with the OS that is installed on the SUT. To locate the MAC address: If the SUT is running Windows* NT*, run ipconfig /all from a DOS shell. If the SUT is running Windows 95 or Windows 98, click Start button, click Run and type winipcfg.

6) From the Test Server, launch the Remote Wake-Up tool (rwu.exe) located in the RWUTool subdirectory (if the default directory was selected during the install, the location will be (c:\Program Files\Intel\WfM_Self_Test\RWUTool).

7) To verify connectivity from the Test Server to the SUT select the Ping Test from the Run Tests menu. Type the IP address of the SUT in the Ping Machine Address field and click OK (if configured, the SUT name can be used instead of the IP address). The results should indicate “Test Passed,” verifying connectivity. If the ping was successful, the IP

address, along with the ICMP data packet size and time stamp information, will be displayed. If ping test fails, verify that a valid, bound IP address exists on the client system. (On a DOS client this implies that a TCPIP driver has been loaded and that a valid bound IP address exsits. Performing the API PXE test before the RWU test is a method that will set up the correct conditions.)

8) Shut down the SUT to its sleep state.

9) On the Remote Wake-Up tool, select Wake-Up Packet Test from the Run Tests menu. Enter the MAC address of the SUT in the MAC Address window (leave the IP Address window empty) and then click OK.

Interpreting the Results of the Remote Wake-Up Test

The SUT should recognize the wake-up packets transmitted from the Remote Wake-Up tool and transition to a fully operational state.

Note: Packets sent by the RWU tool will only work in local network configurations and will not be passed on through routers.

Section

4

Instrumentation Testing

Instrumentation Requirements and Coverage

Requirements	Desktop	Mobile	Server	Self-Test Tool
Management Framework (DMI Version 2.0 Service Provider, SNMP Agent or WBEM Framework) Installed and Active	**Required Either DMI, WBEM or SNMP (See backward compatibility requirements below)**	**Required Either DMI, WBEM or SNMP (See backward compatibility requirements below)**	**Required Either DMI, WBEM or SNMP (See backward compatibility requirements below)**	**DMI: COMPCHK SNMP: SNMPCheck**
Local and Remote Access to Management Data via Standard Access Mechanisms	**Required**	**Required**	**Required**	**DMI: COMPCHK SNMP: SNMPCheck**
Events Generated according to Standard Models (DMI, SNMP or WBEM/CIM)	**Recommended**	**Recommended**	**Recommended**	**DMI: COMPCHK SNMP: SNMPCheck**
DMI Events conform to DMI Event Model	**Required if DMI Events Implemented**	**Required if DMI Events Implemented**	**Required if DMI Events Implemented**	**DCTS tool**
WBEM Events Conform to CIM Event Model	**Future**	**Future**	**Future**	**N/A**
SNMP Traps conform to "DMTF SNMP to DMI Mapping Standard".	**Required if SNMP Framework implemented**	**Required if SNMP Framework implemented**	**Required if SNMP Framework implemented**	**SNMPCheck**
Instrumentation Supports Dynamic Devices	**Required**	**Required**	**Required**	**DMI: COMPCHK SNMP: SNMPCheck**
Instrumentation Deployed and maintained with Product and Platform	**Required --CIM Recommended where supported by platform/OS**	**Required - CIM Recommended where supported by platform/OS**	**Required - CIM Recommended where supported by platform/OS**	**DMI: COMPCHK SNMP: SNMPCheck**

Management Data Available	**Required**	**Required**	**Required**	**DMI: COMPCHK** **SNMP: SNMPCheck**
Backward Compatibility with the WfM 1.1 Specification for Data and Events	**Required: Data and Events Visible via DMI**	**Required: Data and Events Visible via DMI**	**Required: Data and Events Visible via DMI**	**DMI: COMPCHK** **SNMP: SNMPCheck**

DMI: COMPCHK

Overview of COMPCHK test tool

The COMPCHK test tool provides the following capabilities:

- Connects to the DMI Service Provider of the SUT via Remote Procedure Call (RPC).
- Performs a syntax check of the MIF entries contained in the Management Information Format (MIF) database of the SUT by comparing them to the corresponding entries in a master file (master.mif).
- Tests for the presence of certain required groups listed in the appropriate sections of the WfM Baseline v2.0 Specification. This is accomplished by checking for the presence of MIF entries contained in several ".REQ" files (provided with the tool).

Test Procedure

1) Download the master.mif file from the DMTF Web site (http://www.dmtf.org/) and place this file in the \Compchk sub-directory.

2) Ensure that the SUT is booted and that the instrumentation is properly installed. (You can verify that the instrumentation is installed by using a DMI browser to view instrumented attributes.)

3) Find and record the IP address of the SUT. Instructions for locating the IP address can be found in the documentation provided with the OS that is installed on the SUT. The IP address can be found by running IPCONFIG from a DOS shell (if the SUT is running Windows NT), or by running WINIPCFG from the Start menu, Run option (if the SUT is running Windows 95 or Windows 98).

4) On the WfM Test Server, launch the COMPCHK test tool (the default path is c:\Program Files\Intel\WfM_Self_Test\CompChk\compchk2.exe).

5) A window titled DMI 2.0 MIF Conformance Checker will appear on the WfM Test Server.

6) Enlarge the COMPCHK window to its maximum size. Adjust the display dialog box within the window so that the upper section containing the Candidate MIFs, Reference MIFs and Required Groups boxes allows the tester to access the Start button.

7) Click the Service Provider button located below the Candidate MIFs box.

8) Click the New Target button and enter the IP address or name of the SUT in the System Name box (for example, 168.168.125.65). Ensure that the RPC Type is set to DCE and the Transport Type is set to ncacn_ip_tcp; then click OK.

9) If an RPC connection is established to the SUT, you will see the SUT's MIFs listed in the Candidate MIFs box.

10) Go to Reference MIFs list box and click the Add button. Select the master.mif file that you previously downloaded from the DMTF Web site, then click Open. Wait for the contents of the master.mif file to be installed (this may take several minutes).

11) Go to the Required Groups Definition Files list box and click Add button. Select the appropriate platform required group file [desktop2.req, mobile2.req, server2.req], then click Open. These files contain the WfM v2.0 required groups that correspond to the platform type.

12) Go to Required Groups Definition Files list box and click the Add button. Select the appropriate .req file(s) for the platform being tested:

	Desktop	Mobile	Server
Platform Required	Desktop2.req	Mobile2.req	Server2.req
Resource.req	Required if the OS provides Instrumentation	Required if the OS provides Instrumentation	Recommended
Monitor.req	Required if system supports DDC interfaces	Required if system supports DDC interfaces	
Required if Present		M_Network.req M_Slots.req	
Required if supported by the platform			S_Cooling.req S_SHWSec.req S_SPwrCnt.req S_Tprobe.req S_Vprobe.req
Required if provided on the motherboard			S_Network.req S_Ports.req
Required if Instrumented			Masstore.req RAID.req

13) Click Open and wait for the file to be installed.

14) Repeat Steps 12-13 until all required files are added into the test.

15) Click the Start button. A status bar and results should appear in the output window. After the test completes, enlarge the test results window until the vertical scroll bar appears on the results window. You can now interpret the instrumentation test results.

    Note: The results provide both Instrumentation and Required Group checking.

16) Test the SUT for Instrumentation support of Dynamic Devices. Using a DMI browser on the SUT, look at a DMI required group that has a redundant, hot-swappable device. (This device could be a PCMCIA hard drive for a mobile platform, or a hot-swappable disk drive for a server platform, or a USB device.) Check the number of devices installed.

17) Hot-insert an additional device and verify that a new row is dynamically added to the required group (for example, if you hot-swap a disk device, the change should be visible in the Disk group). There may be a delay between device insertion and instrumentation update.

18) Remove the device being tested from the slot. Verify that a row has been deleted from the required group (for example, if a disk device is removed, the change should be visible in the Disk group). There may be a delay between device removal and instrumentation update.

Pass/Fail Criteria

1) **Management Framework & Remote Access**

   Initial remote connection and MIF file download by the WfM Test Server to the SUT verifies support for Remote Procedure Call and the presence of a DMI 2.0 Service Provider.

2) **Standard Group Conformance**

   The first portion of the results determines if the MIFs contained in the instrumentation MIF database match the DMI v2.0 requirements. Any failures are preceded with *****> . All failures must be corrected for the instrumentation layer to pass this test.

3) **WfM 2.0 Required Groups**

   The results of testing for WfM required groups are logged after the results for standard group conformance. There is a separate section for each REQ file that is currently being used for the test. Any missing groups will be flagged. These omissions must be corrected in order to pass the instrumentation test.

   Note: COMPCHK tests for the presence of the groups defined in the accompanying .req files. Errors reported by COMPCHK must be checked to ensure validity since the platform type and system design may impact the level of testing. For example, if a server system is tested using the S_Network.req and fails, then the overall test results will be a Fail if the system implements a network adapter on the motherboard, but will be a Pass if a network add-in card is present, but not on the motherboard. Thus, testers should be aware of the SUT's configuration prior to selecting .req files for testing.

   Important: For RAID Subsystem Required Group testing there are two groups tested, DMTF|Aggregate Physical Extent|001 and DMTF|Aggregate Protected Space Extent|001, but only one may be instrumented. The raid.req will pass a candidate system if one or both of these groups are instrumented. According to the WfM 2.0 Specification, the candidate

system should fail only if they have implemented both, so the tester will need to check for this condition.

4) **Instrumentation Support for Dynamic Devices**

   Instrumentation support for Dynamic Devices should pass for various hot-swappable devices which can be determined using a DMI Browser. Prior to device installation in the system, there should be no instance of instrumentation for the test device. After installation, the device's instrumentation should appear in an appropriate category based upon the device's functionality. Once verified, remove the test device from the system and verify that the device instrumentation is removed from the MIF database.

SNMP: SNMPCheck

Overview of the SNMPCheck Tool

SNMP-based instrumentation is tested via the SNMPCheck Test Tool running on a Windows NT 4.0 (or later) platform with SNMP services installed (ships with Windows NT).

The SNMPCheck Test Tool provides the following capabilities:

- SNMPCheck.exe provides the capability of connecting to the SUT via TCP/IP, and checks the contents of the SUT's Management Information Base (MIB) structures. MIB structures are mapped to their DMI Standard Group counterparts. For WfM 2.0 Required group testing, the test configuration should reflect the platform being tested. The results of the SNMPCheck test can be viewed graphically. Overall test results and test failure descriptions can be saved to a text file upon test completion.
- SNMPCheck.exe also provides the capability of capturing SNMP Traps originating from a specified IP address. Capture of an SNMP Trap provides the user with SNMP OID, originating IP address, Generic Trap ID, Enterprise-specific ID, and variable information. Each captured trap is displayed in the order of reception. To view successive traps, the current trap window must be closed.

Test Procedure: SNMP Instrumentation

1) Ensure Windows* SNMP Services is installed prior to launching the SNMP Tool.

2) Launch the SNMPCheck tool. By default, the SNMPCheck.EXE tool is located in the c:\Program Files\Intel\WfM_Self_Test\SNMPTool directory. Click the Configuration button.

3) Select the appropriate configuration file to perform the test.

   - Desktop Platforms = Desktop2.wfm
   - Mobile Platforms = Mobile2.wfm
   - Server Platforms = Server2.wfm

4) Click the Run button.

   - Test against the address of the SUT.
   - The community must always be "public" for WfM v2.0 MIBs.

- Click OK.

5) The Tool will then run and provide the Groups with a corresponding tree structure in the left window and attribute value information in the right window. An initial results summary shows the number of Warnings and Errors detected during testing.

   Note: The SNMPCheck Tool does not map SNMP OIDs into DMTF attributes.

6) Repeat steps 3 – 5 for each appropriate WfM 2.0 Required Group configuration file:

	Desktop	Mobile	Server
Platform Required	Desktop2.wfm	Mobile2.wfm	Server2.wfm
Resource2.wfm	Required unless the OS provides Instrumentation	Required unless the OS provides Instrumentation	Recommended
Monitor2.wfm	Required if system supports DDC interfaces	Required if system supports DDC interfaces	
Required if Present		M_Net2.wfm M_Slot2.wfm	
Required if supported by the platform			S_RIS.wfm
Required if provided on the motherboard			S_Net2.wfm S_Ports.wfm
Required if Instrumented			Mass2.wfm RAID.wfm Mass_Rec2.wfm tests Recommended

7) The SUT must now be tested for Instrumentation support of Dynamic Devices. Using the SNMPCheck Tool on the Test Server, check a required DMI group representing instances of a redundant, hot-swappable device for the number of devices installed. (This device could be a PCMCIA hard drive for a mobile platform, or a hot-swappable disk drive for a server platform, or an USB device).

8) Hot insert an additional device. Verify that a new row is dynamically added to the required group (for example, if a disk device is used, the change should be visible in the Disk group). This may be accomplished by running the SNMP Test again and looking in the corresponding Group tree for added instrumentation.

9) Remove the device from the slot. Verify that a row has been deleted from the required group (for example, if a disk device is used, the change should be visible in the Disk group). Again, run the SNMPCheck Tool to verify the removal of instrumentation data.

Test Procedure: SNMP Traps

1) Ensure that the SNMP Services on the SUT platform are properly configured:

- Community Name = public
- Trap Destination = IP address of the WfM Test Server

2) On the WfM Test Server Launch the SNMPCheck Tool. The SNMPCheck tool is located in the c:\Program Files\Intel\WfM_Self_Test\SNMPTool directory.

3) Click the Start SNMP Trap Mode button

4) In the window that appears, configure the fields as follows:
   - Enter Network Address = IP address of the SUT
   - Enter SNMP Community = public

5) To check the configuration between the SUT and the WfM Test Server, restart the SUT and ensure that the IP address previously assigned remains the same. If the IP address of the SUT changes, change the Trap Destination address in the SNMPCheck Tool before the SUT's OS begins to load.

   As the OS on the SUT loads, the SNMPCheck Tool should capture the following SNMP Traps:

SNMP Event	Generic Trap ID	Enterprise-specific ID	Variable
.1.3.6.1.4.1.311.1.1.3.1	0	0	
.1.3.6.1.2.1.11.0.7.1990836312	3	0	interfaces.ifTable.ifEntry.ifIndex.1
.1.3.6.1.2.1.11.0.7.1990836312	3	0	interfaces.ifTable.ifEntry.ifIndex.2

   This verifies proper SNMP Service operation on the SUT and the WfM Test Server.

6) Select a system device that corresponds with a DMTF-required group that implements a DMTF Event Generation group (USB devices, PC Cards, Hot-swap drives, etc.)

7) Remove or install the device from the SUT.

8) Check the SNMPCheck Tool for an SNMP Trap. The OID of the trap should correspond to the DMTF Event Generation for which the device is instrumented.

9) For systems that implement Enterprise-specific Event Generations and if the SUT's Instrumentation software enables changing tolerance values, then adjust a tolerance parameter to zero or minimal tolerance levels. (Temperature, voltage, free memory, etc.) This should create a Warning or Error on the SUT. Check the SNMPCheck Tool for a corresponding SNMP Trap with an Enterprise OID and values providing a description of the Warning or Error condition.

Pass/Fail Criteria

1) **Management Framework and Remote Access**

   Connection to the SUT is verified by retrieval of DMI-SNMP Group mappings and any associated data contained within the group's tree.

2) **Instrumentation and Management Data**

   After connection to the SUT and access to the SNMP instrumentation, the DMI Standard Groups should appear in the left window. Groups are expandable with DMI attribute data contained within the tree structure. There is no mapping between DMI attribute names and SNMP data.

3) **WfM Required Groups**

   Initial results provided in the Results window indicate a full Pass or Failure. Any Warnings or Failures must be investigated to determine which group check caused the problem. The tool allows the user to save the results to a file. This file provides the overall results along with a description of the failures.

   Passing groups appear with a green  before to the DMTF Standard Group name.

   Failures appear with a  before the DMTF Standard Group name and indicate that either there is no group present or the group contains no attribute data.

   Important: For RAID Subsystem Required Group testing there are two groups tested, DMTF|Aggregate Physical Extent|001 and DMTF|Aggregate Protected Space Extent|001, but only one may be instrumented. The raid2.wfm will pass either group independently. According to the WfM 2.0 Specification, the candidate system should fail if they have implemented both, so the tester will need to check for this condition.

4) **Dynamic Instrumentation**

   Instrumentation support for Dynamic Devices must pass for various hot-swappable devices which can be determined using the SNMPCheck Tool. Prior to device installation in the system, there must be no instance of instrumentation for the test device. After installation, the device's instrumentation should appear in an appropriate category based upon the device's functionality. Upon removal of the test device from the system, device instrumentation must be removed from the MIB database.

5) **SNMP Traps**

The SUT must generate an SNMP Trap with the proper SNMP OID and any appropriate variable information for DMTF Event Generation-type events. These events would include any Warnings or Errors due to device failures (over-temperature, under-voltage, memory shortage, etc.). Hardware configuration changes may also generate SNMP Traps to inform management software of any system modifications. These hardware changes can affect DMTF Standard Groups and WfM 2.0 Required Groups for that platform. SNMP Trap information should indicate the appropriate DMTF to SNMP OID mapping for that device. Refer to the DMTF's master.mif file for DMTF to SNMP mapping.

Section 5

Remote Network Boot Testing

Remote Network Boot Requirements & Coverage

Requirements	Desktop	Mobile	Server	Self-Test Tool
Preboot Execution Environment	**Required**	**Required if LOM[1] is present.** **Recommended if LAN adapter card present**	**Recommended**	**PXETest**
Boot Integrity Services	**Recommended**	**Recommended**	**Recommended**	
SMBIOS v2.2 or later	**Required**	**Required**	**Required**	**SMBIOWFM**
System Boot Status	**Required**	**Required**	**Required**	**SMBIOWFM**
Remote Lockout	**Required**	**Required**	**Required**	**RLockout**
Platform Event Traps	**Recommended**	**Recommended**	**Recommended**	**SNMPCheck**
Boot Integrity Services	**Recommended**	**Recommended**	**Recommended**	**BISTest1**
SMBIOS data	**Required**	**Required**	**Required**	**SMBIOWFM**

[1] LOM – LAN on Motherboard

Preboot Execution Environment

Overview of PXETest tool

Please review, in detail, the PXE Product Development Kit Instructions contained in the DOCS subdirectory (the default directory is c:\Program Files\Intel\PXE\PDK\docs). This document contains information on creating boot files (this must be done for the test to execute successfully), as well as detailed instructions illustrating the configuration and use of this tool.

Client Test Procedure

1) Find and record the MAC address (sometimes called the adapter or physical address) of the SUT. Instructions for locating the MAC address can be found in the documentation provided with the OS that is installed on the SUT. If the SUT is running Windows* NT*, you

can find the MAC address by running ipconfg/all from a DOS shell. If the SUT is running Windows 95 or Windows 98, go to the Start menu, click Run, type in winipcfg, and click OK.

2) Enable packet analysis capability on the PXE server by doing the following: Launch the PXE configuration Utility on the PXE server. In the left pane, right click the "Boot Server" item under the PXE server that is being configured. Now, select the menu "Configure Packet Analysis" You'll see a dialog box entitled "Packet Analysis". Click the check box "Packet Analysis Test on" and stop and restart the PXE services.

3) Configure the SUT for a Network Boot as described in the PXE PDK Instructions and then Re-boot the SUT.

4) The SUT should begin the remote boot sequence and contact the PXE Server.

5) At the prompt, press F8 for Network Boot Menu.

6) Select APITEST to begin testing. This next phase of testing will take several minutes. When the PXE API Test is complete, a Pass/Fail message will be displayed on the SUT's monitor and the SUT will be left at the a:\> prompt. The results of the test are available on the PXE Server.

Client Pass/Fail Criteria

1) On the PXE Server, look in the Testlog directory (the default is c:\Program Files\Intel\PXE\PDK\Testlog). A subdirectory has been created with a name that consists of the last eight characters of the SUT's MAC address (for example, 000AC96800D4). Go to that subdirectory.

2) Examine the file named TestSum.txt.

   - The top portion of this file contains the results of the analysis of the DHCP packets transmitted by the SUT during the test. Only the first packet needs to be analyzed (the others are not applicable). Any failures noted in the first packet must be corrected. You can find the raw packet associated with the analysis in Cpkt1.txt. The other two Cpkt2.txt and Cpkt3.txt are the DHCP Reply packet and the Boot Server Reply packet.

   - The next portion of this file contains the results of the API tests. Any error in required calls must be corrected to pass this test.

3) Examine the file named ndistest.txt. This file contains the results of performing a series of file transfers between the SUT and the PXE Server. Any errors here may indicate a problem with the dynamic functionality of the Universal NIC Driver Interface (UNDI).

The testlog directory for the SUT (directory labeled with the MAC Address) must be deleted or moved to another location before another test is performed. If this is not done, subsequent test results will be appended to previous results, complicating interpretation. Failure to move or delete this directory before re-testing the SUT may invalidate the instructions in this section.

Bootserver Test Procedure

1) Configure a client system for a Network Boot as described in the PXE PDK Instructions and then Re-boot the client.

2) Add the BootServer under test to the list of available BootServers on the WfM Test Server. (Refer to the PXE PDK Instructions for adding BootServers to the list.)

3) Power on the PXE Client. The PXE client will begin the remote boot sequence and contact the WfM Test Server.

4) At the prompt, press F8 for Network Boot Menu.

5) A list of available BootServers will be presented on the Client's monitor. Select the Bootserver under test from the list of available BootServers. (If the Bootserver under test is not in the list then it was not added correctly to the WfM Test Server.)

Bootserver Pass/Fail Criteria

1) The Bootserver passes the test if the client successfully boots the boot program provided by the Bootserver.

Boot Integrity Services

Overview of BIS self test tool

The BIS self test tool runs on a client platform in a suitably configured DOS environment. It provides the following capabilities:

- Confirms that the proper type-31 SMBIOS entry has been made in the SMBIOS table. This table entry is used by software that calls BIS. It allows the caller to find and bind to the BIS entry point.

- Confirms that BIS can be invoked, started, and shut down.

- Confirms that all BIS functions are present and can be invoked.

- Confirms the normal operation of BIS functions that do not require the caller to generate a digital signature.

The BIS self-test tool does not do a exhaustive test of correct and incorrect parameter combinations of BIS functions. Furthermore, this limited-size self-test does not have the capability to generate digital signatures. Thus tests of some BIS functions are limited to test cases that do not require the caller to generate a digital signature.

A more extensive BIS client software test is available in a BIS SDK supplied by Intel®. This more extensive test covers success and failure scenarios of BIS functions, parameter bounds tests, memory-leak tests, endurance tests, and tests involving a variety of scenarios requiring dynamic generation of digital signatures. This test runs in a client-server configuration where the tests are driven from a Windows NT platform connected to the platform-under-test through

a network. This BIS SDK can be obtained from Intel® at no charge under a limited-time restricted use license agreement. Please contact Intel® for terms of this license.

Test Procedure

1) Boot the client platform to a DOS environment. There are a few details for configuring the DOS environment discussed in the c:\Program Files\Intel\Bis\Local\BISWFMSelfTest document.

2) NOTE: A DOS text version "BISTest.txt" is also available in the is c:\Program Files\Intel\WfM_Self_Test\Bis\Local directory.

3) Invoke "bistest1.exe" from the DOS command prompt. It is available under the c:\Program Files\Intel\WfM_Self_Test\Bis\Local directory. It is also available under the c:\Program Files\Intel\WfM_Self_Test\Boot Disk directory.

4) Inspect the console output from bistest1.exe for the expected output as described in the BISWFMSelfTest document.

Pass/Fail Criteria

The console output from bistest1.exe should be as described in the BISWFMSelfTest document.

Remote Lockout

Overview of Remote Lockout test tool

The Remote Lockout self test tool runs on a client platform in a suitably configured DOS environment. It provides the following capabilities:

- Confirms that the Remote Lockout interface version is V1.10.
- Confirms that the BIOS supports Remote Lockout Interfaces **Inquire Lockout Capabilities** (2500H), the **Set Remote Lockouts** (2502H), and the **Get Remote Lockouts** (2501H.
- Confirms that Remote Lockout interfaces do not use reserved bits.
- Determines which remote lockout features are supported and then tests each interface by validating set and clear functionality.

The Remote Lockout self-test tool does not do an exhaustive test of correct and incorrect parameter combinations of Remote Lockout functions. Furthermore, this limited self-test does not have the capability to verify lockout features due to the manual nature of such testing. Therefore tests are limited to test cases that do not require the user to perform manual intervention.

Test Procedure

1) Boot the client platform to a DOS environment.

2) Invoke “RLockout.exe” from the DOS command prompt. It is available under the c:\Program Files\Intel\WfM_Self_Test\RLockOut directory. It is also available under the c:\Program Files\Intel\WfM_Self_Test\Boot Disk directory.

3) Inspect the console output from the RLockOut.exe tool.

Pass/Fail Criteria

The console output from RlockOut.exe indicates which Remote Lockout features are available and the results of testing each feature. The test summary report must indicate that no errors were detected. A total error count is also provided in the summary.

SMBIOWFM

Overview of SMBIOWFM tool

This tool verifies that a system implements the table convention defined by the System Management BIOS Reference Specification, v2.2 or later, that the structure table is traversable and that no handle is repeated within the table. SMBIOWFM also enables the user to check SMBIOS structures for requirements defined by the Wired for Management Baseline Specification v2.0. The following table lists the structures and data fields checked by the tool:

Structure Type	Structure Data Fields	Data Requirements
Table Entry Point	⇒ Table Anchor string ⇒ SMBIOS Version ⇒ Intermediate Anchor String	⇒ "_SM_" is present in the address range 0xF0000 to 0xFFFFF (16-byte boundary) ⇒ Version 2.2 or later ⇒ "_DMI_"
BIOS Information	⇒ BIOS Version ⇒ BIOS Release Date	⇒ non-null string ⇒ non-null string
System Information	⇒ Manufacturer ⇒ Product Name ⇒ UUID ⇒ Wake-up Type	⇒ non-null string ⇒ non-null string ⇒ Valid UUID Set (non-zero) ⇒ Unknown is not allowed
System Enclosure	⇒ Manufacturer ⇒ Type	⇒ non-null string ⇒ Unknown is not allowed

Processor Information • One Structure per system processor	⇒ Socket Designation ⇒ Processor Type ⇒ Max Speed ⇒ Processor Upgrade ⇒ Status ⇒ Processor Manufacturer ⇒ Processor Family ⇒ Current Speed ⇒ Lx Cache Handle	⇒ non-null string ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Indicates Population (CPU Status sub-field set to 'Known' value if Populated) ⇒ non-null (If Status = Populated) ⇒ Unknown is not allowed (If Status = Populated) ⇒ Unknown is not allowed (If Status = Populated) ⇒ 0xFFFF or references a valid Cache Information Structure
Cache Information • One per external cache	⇒ Socket Designation ⇒ Cache Configuration	⇒ non-null if cache is external to the CPU ⇒ Unknown is not allowed if cache is present (Installed size = non-zero)
System Slots • One structure per upgradeable system slot	⇒ Slot Designation ⇒ Slot Type ⇒ Slot Data Bus Width ⇒ Slot ID ⇒ Slot Characteristics 1 & 2 ⇒ Current Usage	⇒ non-null string ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Set to 'Available' or 'In-use' if presence is detectable (PCI) else set to Unknown (ISA)
Physical Memory Array	⇒ Location ⇒ Use ⇒ Memory Error Correction ⇒ Maximum Capacity ⇒ Number of Memory Devices	⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ non-zero → Identifies the number of Memory Device structures that are associated with this Physical Memory Array.
Memory Device • One structure per socketed system-memory device	⇒ Device Locator ⇒ Memory Array Handle ⇒ Data Width ⇒ Size ⇒ Form Factor ⇒ Device Set ⇒ Total Width	⇒ non-null string ⇒ Contains the handle associated with the Physical Memory Array structure to which this device belongs ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Unknown is not allowed ⇒ Not set to 0xFFFF if Size = non-zero
Memory Array Mapped Address • One structure per contiguous block of memory addresses mapped to a Physical Memory Array	⇒ Ending/Starting Address ⇒ Memory Array Handle ⇒ Partition Width	⇒ Ending Address is larger than Starting Address and each structure's range is unique and non-overlapping ⇒ References a Physical Memory Array structure ⇒ non-zero

Memory Device Mapped Address • Sufficient structures to map memory devices to their memory-array mapped addresses.	⇒ Ending/Starting Address ⇒ Memory Device Handle ⇒ Memory Mapped Address Handle ⇒ Partition Row Position ⇒ Interleave Position ⇒ Interleaved Data Depth	⇒ Ending Address is larger than Starting Address and each structure's range is unique and non-overlapping ⇒ References a Memory Device Structure ⇒ References a Memory Array Mapped Address structure ⇒ Not set to 0, 0xFF, or greater than the Partition Width of the referenced Memory Array Mapped Address structure ⇒ Not set to 0xFF ⇒ Not set to 0xFF
Boot Integrity Services (BIS) • Optional structure - tested if present	⇒ 16-bit real-mode and 32-bit flat protected-mode entry points	⇒ non-zero
System Boot Information	⇒ Reason Code Data	⇒ At least one byte of System Boot Status

Note: All structures are checked for length and checksum.

The SMBIOS Tool runs in the following environment:

- x86-architecture CPU
- DOS 3.3 or later
- The system is operating in real-mode, without v86 extensions

All results are written to the screen with a user option of re-directing the screen output to a file (in other words, SMBIOWFM > smbios.log).

Test Procedure

1) Save the SMBIOWFM Tool to a bootable floppy disk. SMBIOWFM Tool is located in c:\Program Files\Intel\WfM_Self_Test\SMTool. It is also available under the c:\Program Files\Intel\WfM_Self_Test\Boot Disk directory.

2) Insert the disk containing the SMBIOWFM.EXE tool into the SUT and initiate a reboot.

3) At the command prompt, launch the SMBIOWFM tool. You'll see results on the SUT monitor; you can also redirect results to a file (SMBIOWFM > smbios.log).

Pass/Fail Criteria

1) The SMBIOWFM Tool should report no errors.

2) Any reported errors should be analyzed utilizing a detailed test dump.

SNMPCheck

Overview of SNMPCheck tool

Platform Event Traps are tested via the SNMPCheck test tool running on a Windows NT 4.0 (or later) platform with SNMP services installed (ships with Windows NT).

The SNMPCheck Test Tool provides the following capabilities:

- SNMPCheck.exe provides the capability of capturing SNMP Traps originating from a specified IP address. Capture of an SNMP Trap provides the user with SNMP OID, originating IP address, Generic Trap ID, Enterprise-specific ID, and variable information. Each captured trap is displayed in the order of reception. To view successive traps, the current trap window must be closed.

- SNMPCheck.exe also provides the capability of connecting to the SUT via TCP/IP, and checks the contents of the SUT's Management Information Base (MIB) structures. MIB structures are mapped to their DMI Standard Group counterparts. For WfM 2.0 Required group testing, the test configuration should reflect the platform being tested. The results of the SNMPCheck test can be viewed graphically or saved to a text file.

Test Procedure

4) On the WfM Test Server Launch the SNMPCheck Tool.

5) Click the Start SNMP Trap Mode button.

6) In the window that appears, configure the fields as follows:
   - Enter Network Address = IP address of the SUT
   - Enter SNMP Community = public

7) Initiate the boot sequence on the SUT.

8) Monitor the SNMPCheck tool for SNMP Traps. Since the tool filters out all SNMP traps other than the configured IP address, only the SUT will produce an SNMP Trap window.

Pass/Fail Criteria

1) The SNMP Trap PDU format must comply with the Platform Event Trap Format Specification where the enterprise OID = **is0(1).org(3).dod(6).internet(1).private(4).enterprises(1).wired_for_management(3183).PET(1).version(1)**

2) The GUID field in the Variable-Bindings portion of the PDU must provide a valid GUID (non-zero).

Section 6

Problem Resolution Requirement Testing

Problem Resolution Requirements & Coverage

Requirements	Desktop	Mobile	Server	Self-Test Tool
Standard Trouble Ticket Agent Installed and Active	**Recommended**	**Recommended**	**Recommended**	**See Note #1**
SES Compliance Level 2	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**See Note #1**
Transaction Objects	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**See Note #1**
Required Transactions	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**Required If pr01 is met on the platform**	**See Note #1**
Trouble Ticket Initiation	**Recommended**	**Recommended**	**Recommended**	**See Note #1**

Note #1: Read the document entitled PRSTest.doc, which is included in the c:\Program Files\Intel\WFM_SELF_TEST\PRSTool\docs directory if installed. This document provides guidelines for testing Problem Resolution Software implementations.

Note #2: See Section 7 for a complete list of Problem Resolution requirements for desktop, mobile, and server platforms.

Section

7

Platform Checklists

Desktop Platform Checklist

The following table summarizes the WfM Self-Test Toolkit in relation to the Wired for Management Baseline Specification v2.0 for Desktop systems:

Ref #	Capability	WfM v2.0 Required?	Self-Test Tool
in01	Management Framework (DMI Version 2.0 Service Provider, SNMP Agent or WBEM Framework) Installed and Active	Required. Either DMI, WBEM or SNMP	**DMI: COMPCHK** **SNMP: SNMPCheck**
in02	Local and Remote Access to Management Data via Standard Access Mechanisms	Required	**DMI: COMPCHK** **SNMP: SNMPCheck**
in03	Events Generated according to Standard Models (DMI, SNMP or WBEM/CIM)	Recommended	**DMI: COMPCHK** **SNMP: SNMPCheck**
in04	DMI Events conform to DMI Event Model	Required if DMI Events Implemented	**DCTS**
in05	WBEM Events Conform to CIM Event Model	Future	
in06	SNMP Traps conform to “DMTF SNMP to DMI Mapping Standard”.	Required if SNMP Framework implemented	**SNMPCheck**
in07	Instrumentation Supports Dynamic Devices	Required	**DMI: COMPCHK** **SNMP: SNMPCheck**
in08	Instrumentation Deployed and maintained with Product and Platform	Required -- CIM Recommended where supported by platform/OS	**DMI: COMPCHK** **SNMP: SNMPCheck**
in09	Management Data Available	Required per Desktop Checklist	**DMI: COMPCHK** **SNMP: SNMPCheck**

in10	Backward Compatibility with the WfM 1.1 Specification for Data and Events	Required: Data and Events Visible via DMI	**DMI: COMPCHK SNMP: SNMPCheck**
pf01	Preboot Execution Environment	Required	**PXETest**
pf02	SMBIOS 2.2 or later	Required	**SMBIOWFM**
pf03	System boot status	Required	**SMBIOWFM**
pf04	Remote lockout	Required	**RLockout**
pf05	Platform Event Traps	Recommended	**SNMPCheck**
pf06	Boot Integrity Services	Recommended	**BISTEST1**
pf07	SMBIOS data	Required	**SMBIOWFM**
pm01	ACPI-compliant	Required	**ISACPI**
pm02	Recommended Sleep Mode	S3	**N/A**
pm03	Remote Wake-up	Required	**RWUTool**
pm04	Magic Packet	Required	**RWUTool**
pm05	Packet Filtering	Recommended	**RWUTool**
pm06	Wake On Ring	Recommended	
pm07	Bus Power Management	Required	**N/A**
pr01	Standard Trouble Ticket Agent Installed and Active	Recommended	**See Note #1**
pr02	SES Compliance Level 2	Required If pr01 is met on the platform	**See Note #1**
pr03	Transaction Objects	Required If pr01 is met on the platform	**See Note #1**
pr04	Required Transactions	Required If pr01 is met on the platform	**See Note #1**
pr05	Trouble Ticket Initiation	Recommended	**See Note #1**

Note #1: Read the document entitled PRSTest.doc, which is included in the c:\Program Files\Intel\WFM_SELF_TEST\PRSTool\docs directory if installed. This document provides guidelines for testing Problem Resolution Software.

Mobile Platform Checklist

The following table summarizes the WfM Self-Test Tool in relation to the Wired for Management Baseline Specification v2.0 for Mobile systems:

Ref #	Capability	WfM v2.0 Required?	Self-Test Tool
in01	Management Framework (DMI Version 2.0 Service Provider, SNMP Agent or WBEM Framework) Installed and Active	Required. Either DMI, WBEM or SNMP	**DMI: COMPCHK SNMP: SNMPCheck**
in02	Local and Remote Access to Management Data via Standard Access Mechanisms	Required	**DMI: COMPCHK SNMP: SNMPCheck**
in03	Events Generated according to Standard Models (DMI, SNMP or WBEM/CIM)	Recommended	
in04	DMI Events conform to DMI Event Model	Required if DMI Events Implemented	**DCTS**
in05	WBEM Events Conform to CIM Event Model	Future	
in06	SNMP Traps conform to "DMTF SNMP to DMI Mapping Specification".	Required if SNMP Framework implemented	**SNMPCheck**
in07	Instrumentation Supports Dynamic Devices	Required	**DMI: COMPCHK SNMP: SNMPCheck**
in08	Instrumentation Deployed and maintained with Product and Platform	Required -- CIM Recommended where supported by platform/OS	**DMI: COMPCHK SNMP: SNMPCheck**
in09	Management Data Available	Required per Mobile Checklist	**DMI: COMPCHK SNMP: SNMPCheck**
in10	Backward Compatibility with the WfM 1.1 Specification for Data and Events	Required: Data and Events Visible via DMI	**DMI: COMPCHK SNMP: SNMPCheck**
pf01	Preboot Execution Environment	Required if LAN on motherboard present. Recommended if LAN adapter card present.	**PXETest**
pf02	SMBIOS 2.2 or later	Required	**SMBIOWFM**
pf03	System boot status	Required	**SMBIOWFM**
pf04	Remote lockout	Required	**RLockout**
pf05	Platform Event Traps	Recommended	**SNMPCheck**
pf06	Boot Integrity Services	Recommended	**BISTEST1**

pf07	SMBIOS data	Required	**SMBIOWFM**
pm01	ACPI-compliant	Required	**ISACPI**
pm02	Recommended Sleep Mode	S3	**N/A**
pm03	Remote Wake-up	Recommended	**RWUTool**
pm04	Magic Packet	Recommended	**RWUTool**
pm05	Packet Filtering	Recommended	**RWUTool**
pm06	Wake On Ring	Recommended	
pm07	Bus Power Management	Required	**N/A**
pr01	Standard Trouble Ticket Agent Installed and Active	Recommended	**See Note #1**
pr02	SES Compliance Level 2	Required If pr01 is met on the platform	**See Note #1**
pr03	Transaction Objects	Required If pr01 is met on the platform	**See Note #1**
pr04	Required Transactions	Required If pr01 is met on the platform	**See Note #1**
pr05	Trouble Ticket Initiation	Recommended	**See Note #1**

Note #1: Read the document entitled PRSTest.doc, which is included in the c:\Program Files\Intel\WFM_SELF_TEST\PRSTool\docs directory if installed. This document provides guidelines for testing Problem Resolution Software.

Server Platform Checklist

The following table summarizes the WfM Self-Test Tool in relation to the Wired for Management Baseline Specification v2.0 for Server systems:

Ref #	Capability	WfM v2.0 Required?	Self-Test Tool
in01	Management Framework (DMI Version 2.0 Service Provider, SNMP Agent or WBEM Framework) Installed and Active	Required. Either DMI, WBEM or SNMP	**DMI: COMPCHK SNMP: SNMPCheck**
in02	Local and Remote Access to Management Data via Standard Access Mechanisms	Required	**DMI: COMPCHK SNMP: SNMPCheck**
in03	Events Generated according to Standard Models (DMI, SNMP or WBEM/CIM)	Recommended	
in04	DMI Events conform to DMI Event Model	Required if DMI Events Implemented	**DCTS**
in05	WBEM Events Conform to CIM Event Model	Future	
in06	SNMP Traps conform to "DMTF SNMP to DMI Mapping Standard".	Required if SNMP Framework implemented	**SNMPCheck**
in07	Instrumentation Supports Dynamic Devices	Required	**DMI: COMPCHK SNMP: SNMPCheck**
in08	Instrumentation Deployed and maintained with Product and Platform	Required -- CIM Recommended where supported by platform/OS	**DMI: COMPCHK SNMP: SNMPCheck**
in09	Management Data Available	Required per Server Checklist	**DMI: COMPCHK SNMP: SNMPCheck**
in10	Backward Compatibility with the WfM 1.1 Specification for Data and Events	Required: Data and Events Visible via DMI	**DMI: COMPCHK SNMP: SNMPCheck**
pf01	Preboot Execution Environment	Recommended	**PXETest**
pf02	SMBIOS 2.2 or later	Required	**SMBIOWFM**
pf03	System boot status	Required	**SMBIOWFM**
pf04	Remote lockout	Required	**RLockout**
pf05	Platform Event Traps	Recommended	**SNMPCheck**
pf06	Boot Integrity Services	Recommended	**BISTEST1**
pf07	SMBIOS data	Required	**SMBIOWFM**

pm01	ACPI-compliant	Required as defined in [HDG]	**ISACPI**
pm02	Recommended Sleep Mode	S1	**N/A**
pm03	Remote Wake-up	Recommended	**RWUTool**
pm04	Magic Packet	Recommended	**RWUTool**
pm05	Packet Filtering	Recommended	**RWUTool**
pm06	Wake On Ring	Recommended	
pm07	Bus Power Management	Recommended	**N/A**
pr01	Standard Trouble Ticket Agent Installed and Active	Recommended	**See Note #1**
pr02	SES Compliance Level 2	Required If pr01 is met on the platform	**See Note #1**
pr03	Transaction Objects	Required If pr01 is met on the platform	**See Note #1**
pr04	Required Transactions	Required If pr01 is met on the platform	**See Note #1**
pr05	Trouble Ticket Initiation	Recommended	**See Note #1**

Note #1: Read the document entitled PRSTest.doc, which is included in the c:\Program Files\Intel\WFM_SELF_TEST\PRSTool\docs directory if installed. This document provides guidelines for testing Problem Resolution Software.